

MATLAB

Lecturer : Dr. Rafel Hekmat Hameed

University of Babylon

Subject : MATLAB

College of Engineering

Year : Second B.Sc.

Mechanical Engineering Dep.

PLOT

In this section we'll cover

- Creating and labeling two dimensional plots
- Adjusting the appearance of your plots
- Creating multiple figures
- Using subplots
- Creating three dimensional plots
- Using the interactive plotting tools

Two Dimensional Plots

- The xy plot is the most commonly used plot by engineers
- The independent variable is usually called x
- The dependent variable is usually called y

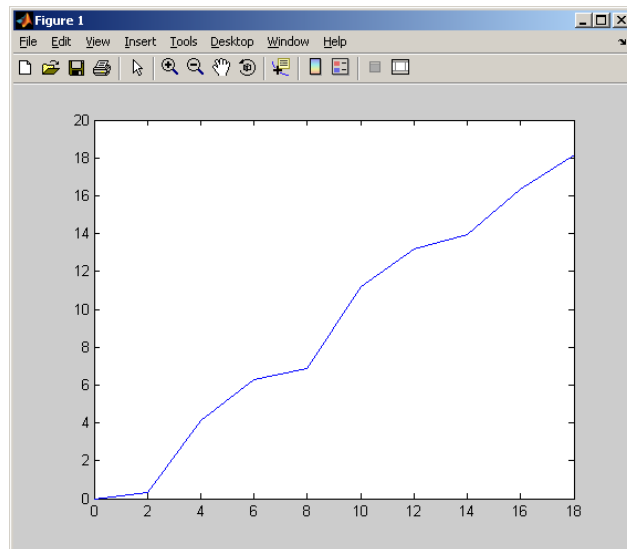
Consider this xy data

time, sec	Distance, ft
0	0
2	0.33
4	4.13
6	6.29
8	6.85
10	11.19
12	13.19
14	13.96
16	16.33
18	18.17

Define x and y and call the plot function

```
Command Window
File Edit Debug Desktop Window Help

>> x=[0:2:18];
>> y = [0, 0.33, 4.13, 6.29, 6.85, 11.19, ...
        13.19, 13.96, 16.33, 18.17];
>> plot(x,y)
```

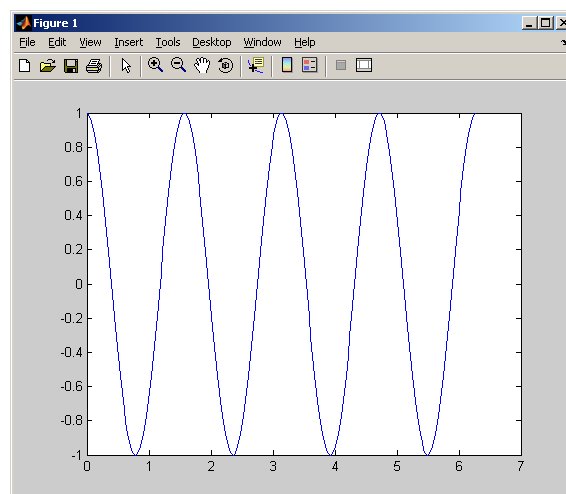


Plots with multiple lines

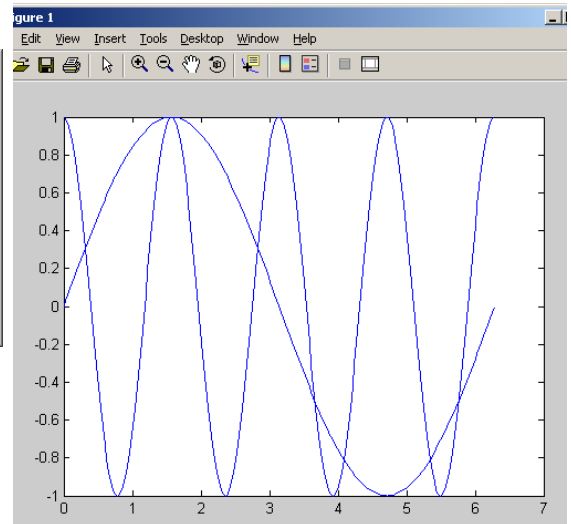
- **hold on**
Freezes the current plot, so that an additional plot can be overlaid
- When you use this approach the additional line is drawn in blue – the default drawing color

```
Command Window
File Edit Debug Desktop Window Help

>> x = 0:pi/100:2*pi;
>> y1 = cos(x*4);
>> plot(x,y1)
>>
```



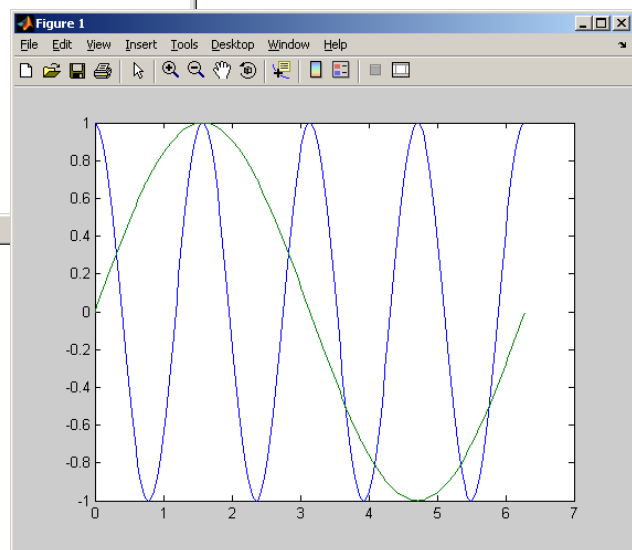
```
Command Window
File Edit Debug Desktop Window Help
>> x = 0:pi/100:2*pi;
>> y1 = cos(x*4);
>> plot(x,y1)
>> hold on
>> y2 = sin(x);
>> plot(x, y2)
```



The second line is drawn on top of the original plot. To unfreeze the plot use the **hold off** command

You can also create multiple lines on a single graph with one command

```
Command Window
File Edit Debug Desktop Window Help
>> x = 0:pi/100:2*pi;
>> y1 = cos(x*4);
>> y2 = sin(x);
>> plot(x,y1,x,y2)
>>
```

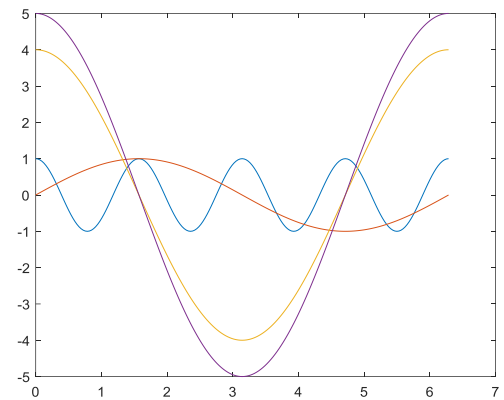


Each set of ordered pairs will produce a new line

If you want to create multiple plots, all with the same x value you can...

- Use alternating sets of ordered pairs
`plot(x,y1,x,y2,x,y3,x,y4)`
- Or group the y values into a matrix
`z=[y1,y2,y3,y4]`
`plot(x,z)`

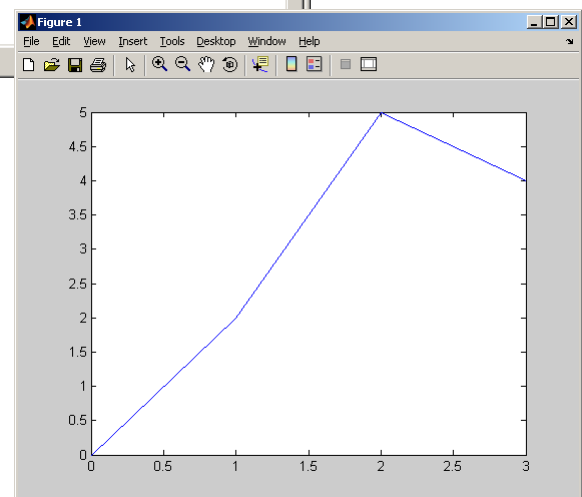
```
Command Window
File Edit Debug Desktop Window Help
>> Y3 = cos(X)*4;
>> Y4 = cos(X)*5;
>> plot(X, Y1, X, Y2, X, Y3, X, Y4)
>> Z=[Y1; Y2; Y3; Y4];
>> plot(X, Z)
>>
```



Plots of Complex Arrays

If the input to the plot command is a single array of complex numbers, MATLAB plots the real component on the x-axis and the imaginary component on the y-axis

```
Command Window
File Edit Debug Desktop Window Help
>> A=[0+0i,1+2i, 2+5i, 3+4i];
>> plot(A)
```



Line, Color and Mark Style

You can change the appearance of your plots by selecting user defined

- line styles
- color
- mark styles

Available choices

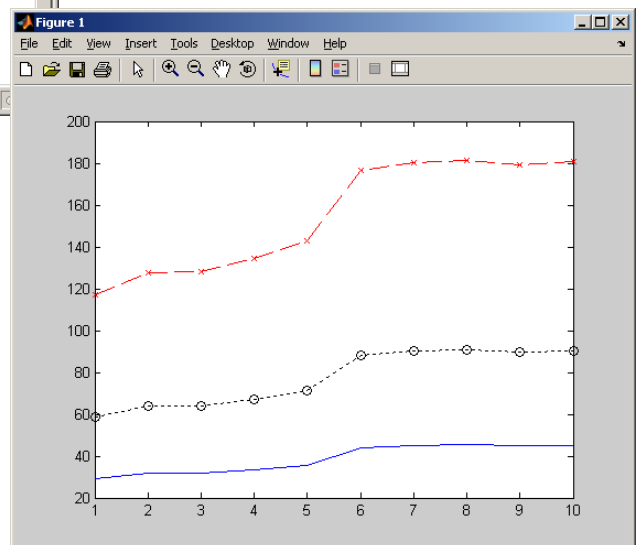
Table: Line, Mark and Color Options

Line Type	Indicator	Point Type	Indicator	Color	Indicator
solid	-	point	.	blue	b
dotted	:	circle	o	green	g
dash-dot	-.	x-mark	x	red	r
dashed	--	plus	+	cyan	c
		star	*	magenta	m
		square	s	yellow	y
		diamond	d	black	k
		triangle down	v		
		triangle up	^		
		triangle left	<		
		triangle right	>		
		pentagram	p		
		hexagram	h		

```

Command Window
File Edit Debug Desktop Window Help
>> x = [1:10];
>> y = [ 58.5, 63.8, 64.2, 67.3, 71.5, 88.3,...
        90.1, 90.6, 89.5, 90.4];
>> plot(x,y,':ok',x,y*2,'--xr',x,y/2,'-b')
>>

```



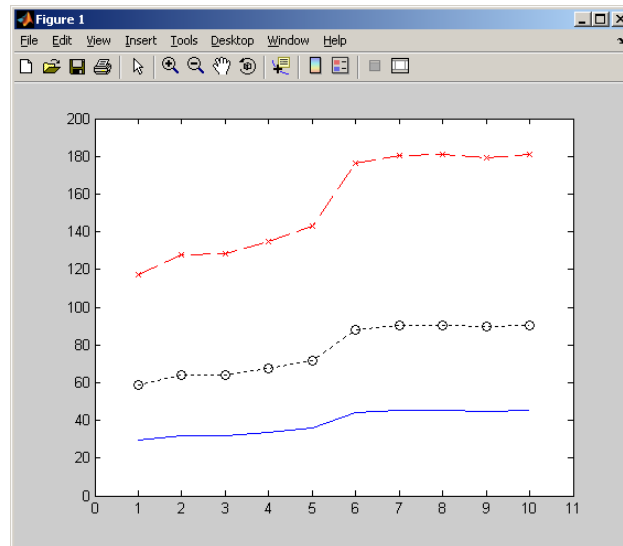
Axis scaling

- MATLAB automatically scales each plot to completely fill the graph
- If you want to specify a different axis – use the axis command
axis ([xmin, xmax, ymin, ymax])

```

Command Window
File Edit Debug Desktop Window Help
>> x = [1:10];
>> y = [ 58.5, 63.8, 64.2, 67.3, 71.5, 88.3,...
        90.1, 90.6, 89.5, 90.4];
>> plot(x,y,':ok',x,y*2,'--xr',x,y/2,'-b')
>> axis([0,11,0,200])
>>

```



Annotating Your Plots

- You can also add
 - legends
 - textbox
- Of course you should always add
 - title
 - axis labels

```

>> x = [1:10];
>> y = [ 58.5, 63.8, 64.2, 67.3, 71.5, 88.3,...
        90.1, 90.6, 89.5, 90.4];
>> plot(x,y,':ok',x,y*2,'--xr',x,y/2,'-b')
>> axis([0,11,0,200])
>> legend('line 1', 'line 2', 'line3')
>> text(1,100,'Label plots with the text command')
>> xlabel('My x label'), ylabel('My y label')
>> title('Example graph for Chapter 5')
  
```

